

Session Code: DAT300

data systems



Overview of what's new for developers in SQL Server "Yukon"

James Hamilton
General Manager
Microsoft Corporation

PDC⁰³

Make the connection

Microsoft[®]

Agenda

- Database Programming Evolution
- .NET Framework Integration
- Data Access and Programmability
- XML Support
- Web Services Support
- New Application Frameworks
- Summary

Early Database Programmability

- Early DB servers were non-extensible
 - Fixed set of types w/o server-side code
- Stored Procs began extensible DB revolution in late 80's
 - Financial community was quick to adopt
- Why stored procedures?
 - Abstraction layer
 - Application specific abstractions
 - Business logic & rich constraint enforcement
 - Security on application abstractions
 - Data intensive code closer to data
- Stored procedures heavily used
- Stored proc languages functionally weak, proprietary, with poor dev tool support

Database Programmability Directions

- Server-side DB programming trends
 - Data-intensive code still is run close to data
 - Symmetric programming language
 - Modern languages with high quality tools
- Leverage programming language community progress
 - Multiple languages & rich development tools
 - Abstract data types, Inheritance, exception handling, encapsulation, ..
 - Rich libraries & application frameworks

Agenda

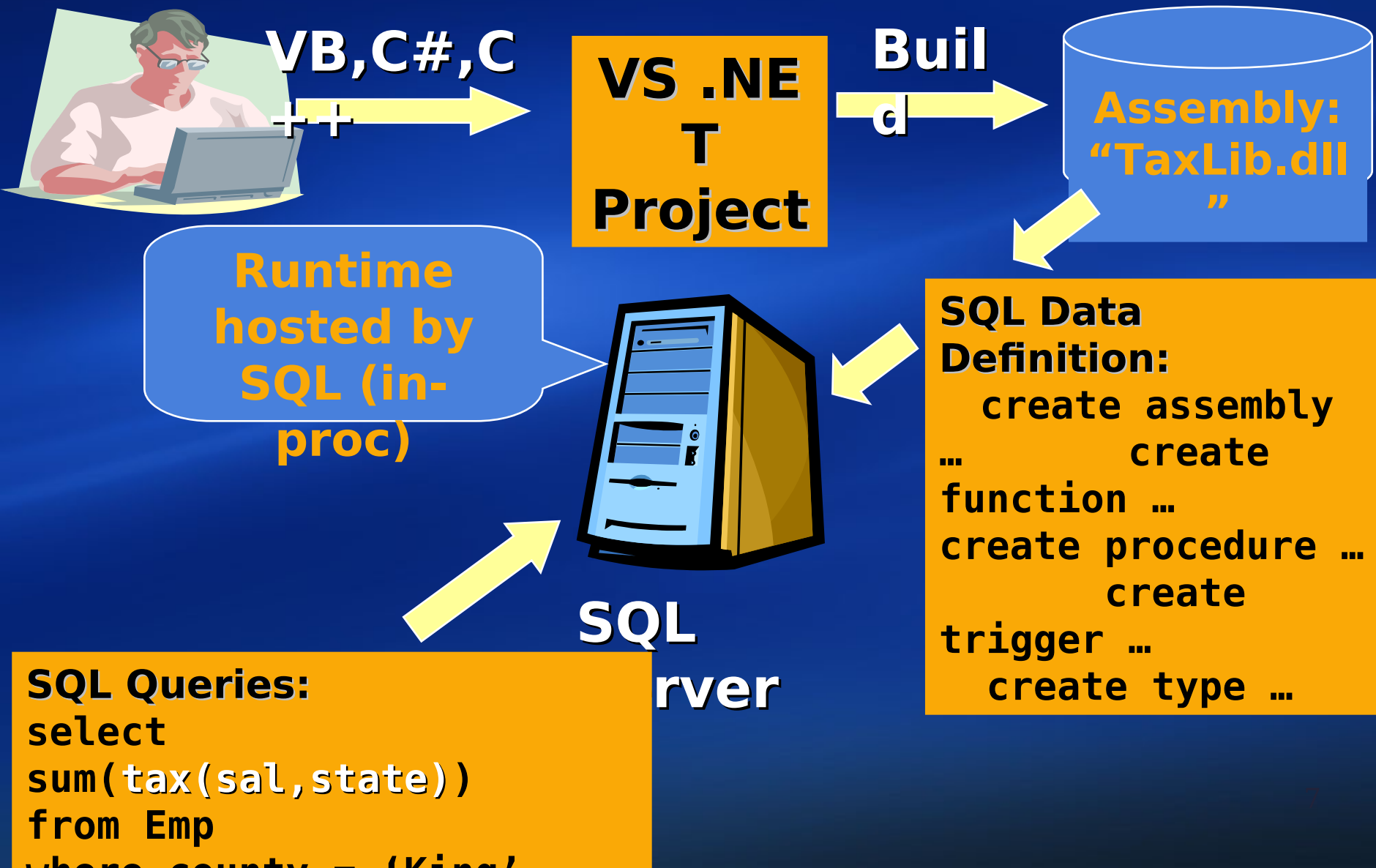
- Database Programming Evolution
- .NET Framework Integration
- Data Access and Programmability
- XML Support
- Web Services Support
- New Application Frameworks
- Summary

.NET Framework Integration

Key Features

- Server-side programming environment for:
 - User Defined Functions, Stored Procedures, Triggers
 - User Defined Types, user defined Aggregates
- In-Proc Data Access (ADO.NET V2 - Whidbey)
- Common ADO .NET Programming Model
 - Both Mid-tier/data tier
- Security
 - Integration of SQL and CLR security
 - Three levels of code access security
 - Safe, External-Access (verifiable), Unsafe
- Tight integration with Visual Studio
 - Authoring, debugging, deployment, & profiling

The Developer Experience



Authoring/Debugging/Deploying

- New Visual Studio project type in “Whidbey” for “Yukon” managed code
- Server debug integration
 - Full debugger visibility
 - Set breakpoints anywhere
- Single step support:
 - Between languages: T-SQL, C#, VB, & C++
 - Between deployment tiers:
 - E.g. ASP.NET, through SQL Server stored proc call, & back to mid-tier

“Yukon” and the .NET Framework

demo

Tom Rizzo
Director
SQL Server

PDC⁰³

Make the connection

.NET Integration

Key Theme: Choice & Control

- Choice of where to run logic
 - Database, for logic that runs close to data
 - Mid-tier, for logic that scales out
 - Symmetric programming model
 - Leverage skills mid-tier & server
- Safe extended stored proc replacement
- Choice of programming language
 - C#, VB.NET, & Managed C++, for a safe, modern execution environment
 - T-SQL enhancements continue
 - Right choice for data-intensive procedures

Agenda

- Database Programming Evolution
- .NET Framework Integration
- Data Access and Programmability
- XML Support
- Web Services Support
- New Application Frameworks
- Summary

TSQL Enhancements

- Data types:
 - Varchar(max) & XML
- Recursive Queries (ANSI standard syntax)
- Exception Handling
 - Handles transaction abort (catch)
- Statement-level recompile
 - Fewer recompiles & less costly
- DDL Triggers
- Queuing Primitives

Data Access

API Enhancements: ADO .NET V2

- Multiple active result sets (MARS)
- Object Persistence FX (ObjectSpaces)
- Query notifications – for cache invalidation
- Server cursors (SqlResultSet)
- Asynchronous client access
- Bulk update, paging, and batching
- XML Data type support
- User Defined Type (UDT) support

Full-text Enhancements

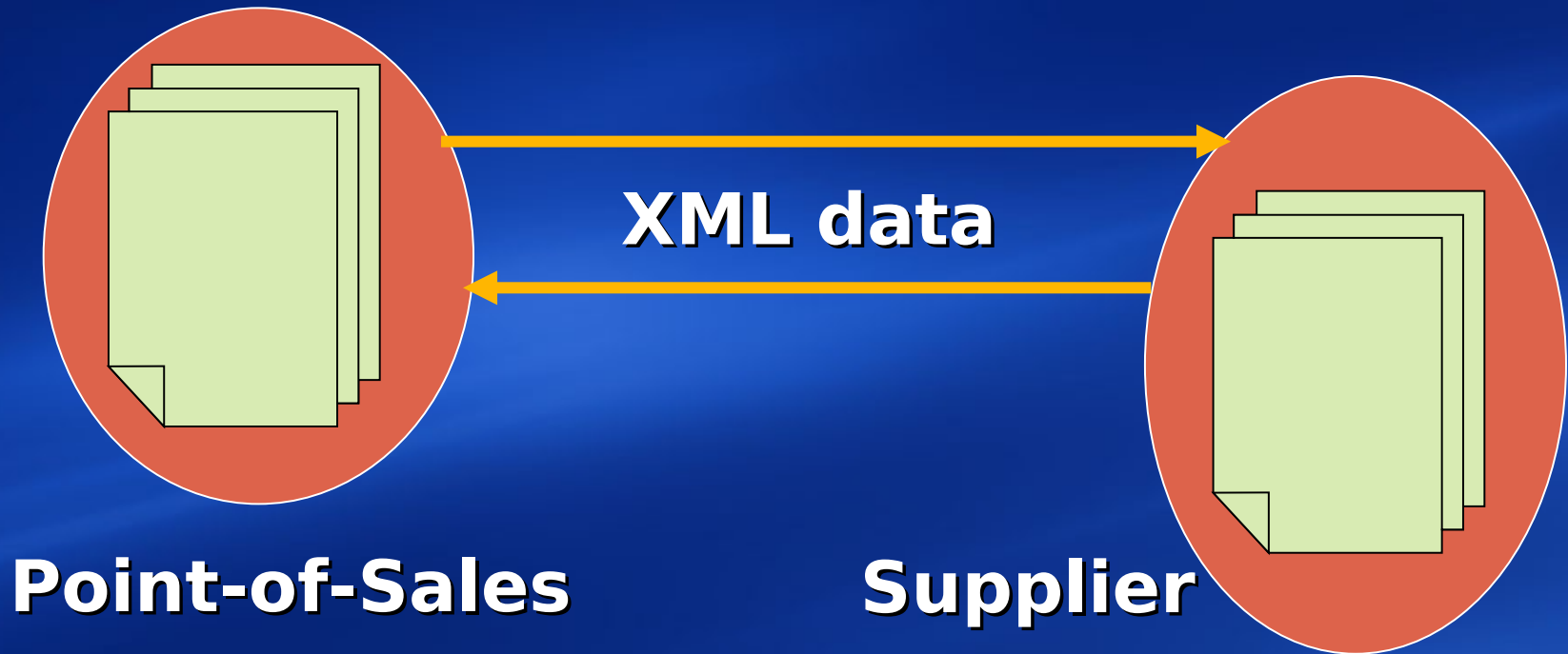
- **Performance & Scalability**
 - Index and Query Performance
 - Tested to over 2,000,000,000 rows
 - Index scales near linearly with data
- **Administrative integration**
 - Transportability via database attach/detach
 - Integrated backup, restore & recovery
 - Full-Text DDL for easier administration
- **Functionality**
 - Thesaurus
 - Diacritic sensitivity/insensitivity
 - Multi-column full-text queries supported
 - `CONTAINS((col1,col2), 'Yukon')`
 - Support for XML data type

Agenda

- Database Programming Evolution
- .NET Framework Integration
- Data Access and Programmability
- XML Support
- Web Services Support
- New Application Frameworks
- Summary

XML Scenarios

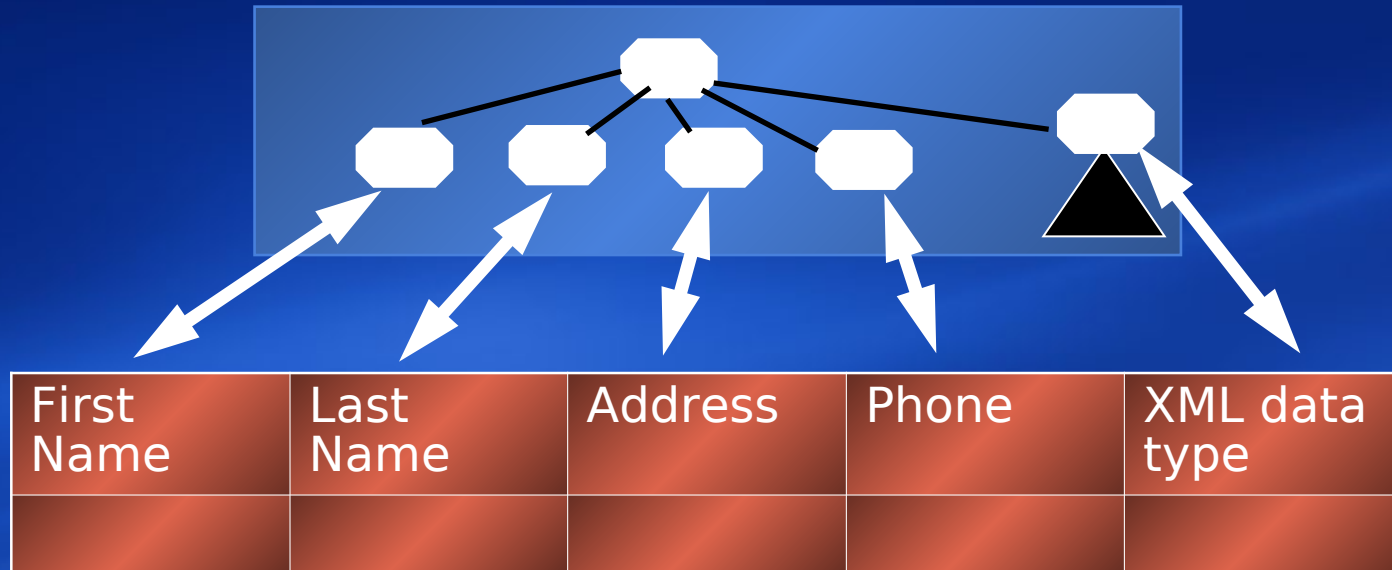
Data Exchange...



Platform independent transport format
Loosely-coupled systems
B2B, B2C, work flow, ...

XML Scenarios

Semi-structured storage...



- XML Datatype
 - Loosely structured data
 - Data with a dynamic schema
- XML Views
 - Mixed data – structured/unstructured
 - XML stores w/o relational support challenged

Native XML Store

XML Data Type

- XML data type
 - Native SQL type
 - Use for column, variable or parameter

```
CREATE TABLE docs (id INT PRIMARY KEY, xDoc  
XML NOT NULL)
```

- Store un-typed or typed XML instances
- Well-formed and validation checks
- Optional XML Schema enforcement

Native XML Store

XML Index

- Create XML index on XML column

```
CREATE XML INDEX idx_1 ON docs (xDoc)
```

- Creates indexes on tags, values & paths
- Speeds up queries
 - Entire query is optimized
 - Same industry leading cost based optimizer
 - Indexes are used as available

Native XML Store

XML Query

- XQuery: query XML documents & data
 - Standards-based: W3C working draft
- In document 123, return section heading of section 3 and later

```
SELECT id, xDoc::query('for $i in /doc[@id = 123]//sec[@num  
>= 3] return <topic>{data($i/heading)}</topic>')  
  
FROM docs
```

1	@id=123
3	@id=30

⇒

1	<topic>Sort</topic> <topic>Search</topic>
3	NULL

“Yukon” and XML demo

Tom Rizzo
Director
SQL Server

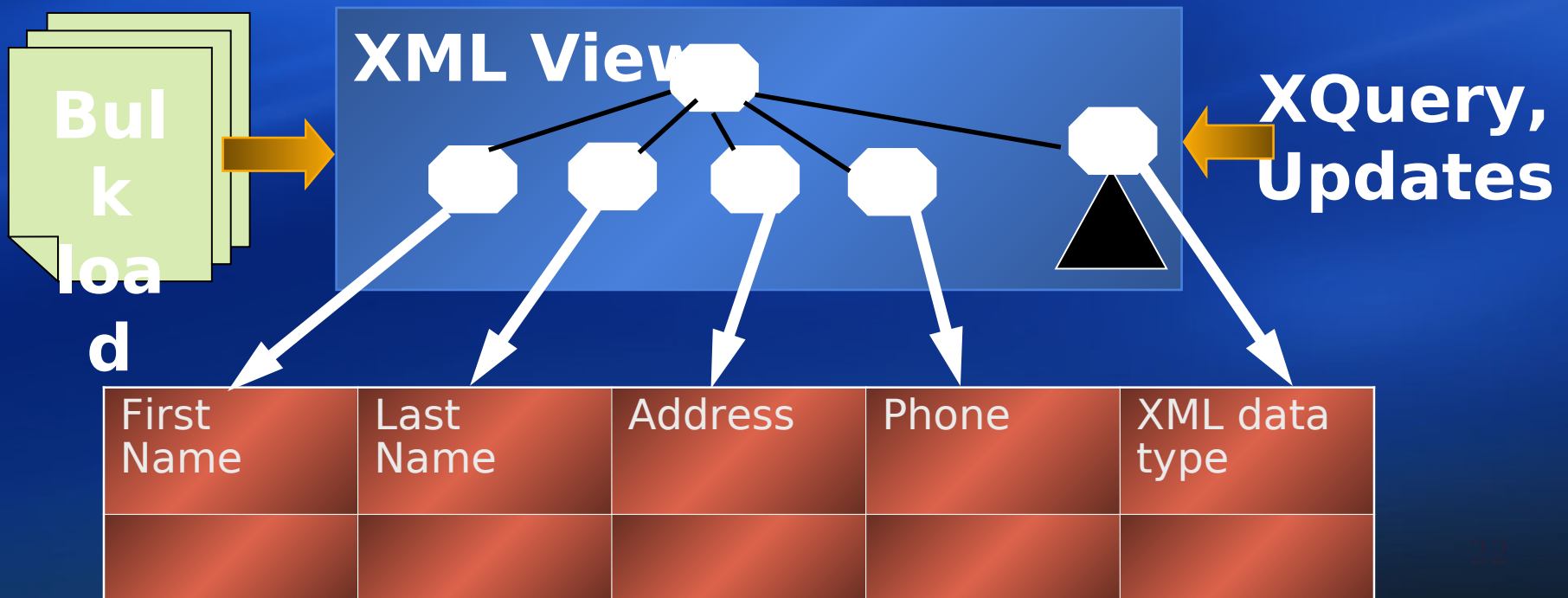
PDC⁰³

Make the connection

XML Views

Overview

- Default XML view of relational data
- User-defined XML views
 - Specified using schema mapping
- Decouples mapping from domain specific schemas



XML Programming Platform

- Rich APIs, tools
 - XMLAdapter, XMLReader, XMLWriter, Schema validation, and XQuery support
 - XQuery builder
 - XML Editor
- Support for new standards
 - XPath 2.0
- High perf XML Parsing & XSLT engines
- XML data type supported in ADO.NET
- XML Datatype/XQuery (DAT402, Wed 10:00 152/153)
- Yukon and SOAP (DAT407, Wed 5:00 515AB)

Agenda

- Database Programming Evolution
- .NET Framework Integration
- Data Access and Programmability
- XML Support
- **Web Services Support**
- New Application Frameworks
- Summary

HTTP/SOAP Features

- SQL & stored proc calls via HTTP/SOAP
 - Easy, standards based connectivity from Unix platforms to SQL Server
- Provides native HTTP listening
 - HTTP endpoint specifying URL, port, reqs
 - Publish WSDL for endpoints
- Standard-based
 - SOAP 1.1 and 1.2, WSDL 1.1, inlined XSD
- Windows and SQL authentication (SSL only)
- Stored Proc can return result as DataSet
 - Compatible with mid-tier programming model

“Yukon” and Web Services

demo

Tom Rizzo
Director
SQL Server

PDC⁰³

Make the connection

Agenda

- Database Programming Evolution
- .NET Framework Integration
- Data Access and Programmability
- XML Support
- Web Services Support
- **New Application Frameworks**
 - SQL Service Broker
 - SQL Server Notification Services
 - SQL Server Reporting Services
- Summary

New Application Frameworks

- Service Broker

- Asynchronous, loosely coupled programming pattern
- Support for queues & service processes

- Notification Services

- High scale event based notification system
- SQL Server, File system, or custom event providers
- Delivery: Email, .Net alerts, file, HTTP, custom distributors

- Reporting Services

- Rich data services no longer an extra cost component
- Sources: SQL, Oracle, XML/A, ODBC, OLE/DB

Summary

- SQL Server .Net Integration
 - Rich, multi-lingual programming model
 - Industry leading development tools
- Improved ADO.NET data access stack
 - Deep server-side storage services
- Integrated Web Services support
- Three New application frameworks
 - Service Broker
 - Notification Services
 - Reporting Services



PDC⁰³

Make the connection

Microsoft Professional Developers Conference 2003

October 26 - 30, 2003, Los Angeles, CA

Microsoft®